

비트 인터리버 기반 극 부호의 개선된 연속적 제거 반전 복호

김하성*, 이현지*, 박호성^o

Improved Successive Cancellation Flip Decoding of Polar Codes Based on Bit Interleaver

Haseong Kim*, Hyunjee Lee*, Hosung Park^o

요약

본 논문에서는 기존 연속적 제거 반전 복호보다 계산 복잡도를 감소시키기 위해서 비트 인터리버 기반 연속적 제거 반전 복호를 제안한다. 또한, 반전될 비트를 선택하는데 밀도 진화 확률을 사용한다. 높은 E_b/N_0 에 대하여, 제안하는 복호는 기존 복호와 유사한 오류 정정 성능을 가지면서, 오류가 발생한 프레임에 대해서 기존 복호보다 최소 35% 감소한 계산 복잡도를 가진다.

Key Words : error correcting codes, polar codes, successive cancellation flip (SCF) decoding, interleaver, density evolution probability

ABSTRACT

In this paper, we propose an improved successive cancellation flip (SCF) decoding algorithm based on bit interleaver to reduce computational complexity compared to the conventional SCF decoding. In addition, we use density evolution probability to select the information bits to flip. The computational complexity of the proposed SCF decoding is less by at least 35% than the conventional one for error frames at a high

E_b/N_0 while keeping a similar error-correction performance to the conventional one.

I. 서론

채널 양극화 현상을 사용하는 극 부호^{[1][2]}는 유한한 코드 길이에서 오류 정정 성능이 좋지 않은 연속적 제거 (successive cancellation: SC) 복호를 대신하기 위해 리스트 개수만큼 병렬 경로를 사용하여 향상된 오류 정정 성능을 가지는 연속적 제거 리스트 (SC list: SCL) 복호를 제안했다^[3]. 하지만 리스트 개수가 증가할수록 높은 하드웨어 복잡도를 갖는 SCL 복호는 IoT 기기처럼 자원이 제한된 장치에서 사용하기 어렵다. 이를 해결하기 위해서 제안된 연속적 제거 반전 (SC flip: SCF) 복호^[4]는 SC 복호보다 향상된 오류 정정 성능과 SCL 복호보다 낮은 하드웨어 복잡도를 가진다. 하지만 SCF 복호는 SC 복호 대비 추가적인 계산 복잡도를 가지는데, 이를 최소화시키기 위해 고정적인 비트만을 선택하여 반전시키는 연속적 제거 반전 (SCF fixed index selection: SCF-FIS) 복호가 제안되었다^[5].

본 논문에서는 기존 SCF 복호의 계산 복잡도를 더욱 감소시키기 위해서 인터리버 기반 연속적 제거 반전 (interleaver-based SCF: ISCF) 복호 알고리즘을 제안한다. 또한, 반전시킬 비트를 선택하기 위해 밀도 진화 확률을 이용한다. ISCF 복호는 인터리버를 통해 재정렬된 순환 중복 검사 (cyclic redundancy check: CRC) 비트를 체크포인트로 사용하여, 모든 정보 비트들을 복호 하지 않더라도 오류 여부를 미리 판단함으로써 계산 복잡도를 줄인다. 높은 E_b/N_0 에서 오류가 발생한 프레임에 대해서만 진행한 시뮬레이션 결과를 통해서, ISCF 복호의 오류 정정 성능은 SCF-FIS 복호와 유사하고, SCF-FIS 복호보다 약 35% 감소한 계산 복잡도를 갖는 것을 확인할 수 있다.

II. 연속적 제거 복호 알고리즘

메모리 자원이 한정적인 장치에서 오류 정정 성능이 향상된 극 부호를 사용하기 위해서 SCF 복호가 제

* 본 연구는 2018년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임 (No. NRF-2018R1D1A1B07051108).

• First Author : Department of LG Smart Hybrid Engineering, Chonnam National University, nivell@nate.com, 학생회원

o Corresponding Author : (ORCID:0000-0001-7854-7792) School of Electronics and Computer Engineering, Chonnam National University, hpark1@jnu.ac.kr, 중신회원

* Department of LG Smart Hybrid Engineering, Chonnam National University, hglee0306@naver.com, 학생회원

논문번호 : 201810-322-A-LU, Received October 12, 2018; Revised November 1, 2018; Accepted December 3, 2018

안되었다⁴⁾. 기본적으로 SC 복호는 입력 비트 \hat{v}_i 을 채널 출력값과 이전까지 추정된 입력 벡터 \hat{v}_1^{i-1} 의 전이 확률을 이용하여 계산된 로그 우도 비 (log-likelihood ratio: LLR)를 사용하여 추정한다. 입력 비트들의 개수는 정보 비트들의 개수 K 와 CRC 비트들의 개수 C 의 합이고, 입력 벡터 v_1^{K+C} 는 수식 (1)로 표현된다.

$$v_1^{K+C} = [v_1, v_2, \dots, v_K, v_{K+1}, \dots, v_{K+C}] \quad (1)$$

SC 복호는 입력 벡터 v_1^{K+C} 를 차례대로 복호하기 때문에 오류 전파특성이 있는데, 오류 전파란 입력 비트 \hat{v}_i 이 잘못 추정되면 이후에 추정되는 입력 비트들도 잘못 추정되는 현상이다. SCF 복호는 채널 잡음에 의해 처음으로 잘못 추정되어 오류 전파를 발생시킬 확률이 높은 입력 비트의 채널 출력값을 반전시켜 향상된 오류 정정 성능을 갖는 기법이다. LLR 기준으로 비트 반전 후보 집합을 정의하는 SCF 복호는 선택 과정에서 추가적인 정렬 및 비교 연산이 필요하고, 이 연산들은 SCF 복호의 계산 복잡도를 증가시킨다.

비트 반전 후보 집합 선택 기준을 변경하여 계산 복잡도를 감소시키는 기법으로서 SCF-FIS 복호가 제안되었다⁵⁾. SCF-FIS 복호의 선택 기준은 오프라인 시뮬레이션을 통해 정보 비트마다 채널 잡음에 의해 처음으로 잘못 추정된 경우의 빈도수이며, 빈도수가 높은 정보 비트들은 후보로 선택된다.

III. 인터리버 기반 연속적 제거 반전 복호

IoT 장치에서 극 부호를 사용하기 위해서는 여전히 기존 SCF 복호의 계산 복잡도를 감소시킬 필요가 있다. 그래서 본 논문에서는 ISCF 복호를 제안한다. ISCF 복호는 먼저 송신기에서 각 CRC 비트가 특정 정보 비트들의 XOR 연산으로 구성되어있다는 점을 이용하여, 각 CRC 비트 앞에 해당 CRC 비트와 관련된 정보 비트들이 존재할 수 있도록 인터리버를 설계하고, 이 인터리버를 사용해서 입력 벡터를 재정렬한다. 수신기 역시 송신기와 같은 CRC 다항식과 인터리버를 사용하기 때문에 재정렬된 CRC 비트 위치를 알고 있다. 인터리버($I(\cdot)$)를 사용하여 재정렬된 입력 벡터 u_1^{K+C} 는 수식 (2)와 같이 표현된다.

$$u_1^{K+C} = I(v_1^{K+C}) \quad (2)$$

알고리즘 1. ISCF 복호 알고리즘
Algorithm 1. The ISCF decoding algorithm

<p>■ 입력 : 채널 출력값(y_1^{K+C}), 인터리버($I(\cdot)$), CRC 비트마다 사전에 결정된 비트 반전 후보 수열 벡터($F_j, j=1, \dots, C$), 최대 후보군 반전 개수(T)</p>
<p>■ 출력 : 추정된 입력 벡터 ($\hat{u}_1^{K+C} = [\hat{u}_1, \hat{u}_2, \dots, \hat{u}_{K+C}]$)</p>
<ol style="list-style-type: none"> 1. y_1^{K+C}로부터 SC 복호를 사용하여, \hat{u}_1^{K+C}을 한 비트씩 추정. 2. 1의 과정에서 현재 추정되는 비트가 CRC 비트에 해당한다면 CRC 검사를 실시. <ol style="list-style-type: none"> 2-1. CRC 검사가 실패하면, 3으로 이동. 2-2. 모든 CRC 검사가 모두 성공이면, 8로 이동. 3. CRC 검사가 실패한 CRC 비트(j)에 따라서 F_j를 선택. 4. F_j중 확률이 높은 1개의 비트 반전 후보를 선택. <ol style="list-style-type: none"> 4-1. 7-1과정을 통해 4과정으로 다시 온 경우, 한 번 선택된 후보들은 제외하고 선택. 5. 후보 비트 위치에 해당하는 y_1^{K+C}값을 반전. 6. 처음부터 다시 SC 복호 수행하여 \hat{u}_1^{K+C} 추정. 7. 6의 과정에서 현재 추정되는 비트가 CRC 비트에 해당한다면 CRC 검사를 실시. <ol style="list-style-type: none"> 7-1. CRC 검사가 실패하면, 4로 이동. 최대 T만큼 반복 가능. 7-2. 모든 CRC 검사가 모두 성공이면, 8로 이동. 8. \hat{u}_1^{K+C} 추정 완료.

ISCF 복호는 두 단계로 구성된다. 첫 번째 단계는 재정렬된 CRC 비트들을 체크포인트로 활용한 SC 복호 과정이다. SC 복호를 통해 u_1^{K+C} 을 차례대로 추정하는데, 추정되는 비트가 CRC 비트라면 CRC 검사를 수행한다. 이때 CRC 검사가 성공이면 계속해서 SC 복호를 수행하여 다음 비트들을 추정하고, 그렇지 않다면 SC 복호를 중지하고 두 번째 단계를 수행한다.

두 번째 단계는 채널 잡음에 의해 처음으로 잘못 추정되었다고 판단되는 정보 비트의 후보들을 반전시키는 과정이다. 첫 번째 단계에서 CRC 검사가 실패한 CRC 비트와 해당 CRC 비트 앞에 존재하는 정보 비트들 사이에서 채널 잡음에 의한 첫 번째 오류 비트가 있다고 판단하고 비트 반전 후보들을 선택한다. 사전 오프라인 밀도 진화 시뮬레이션을 통해서 CRC 비트마다 F_j 가 별도로 정의되어있고, F_j 의 개수는 각 CRC 비트 앞에 존재하는 정보 비트들의 수만큼 존재한다. 이후 선택된 F_j 에 존재하는 비트 위치의 채널 출력값

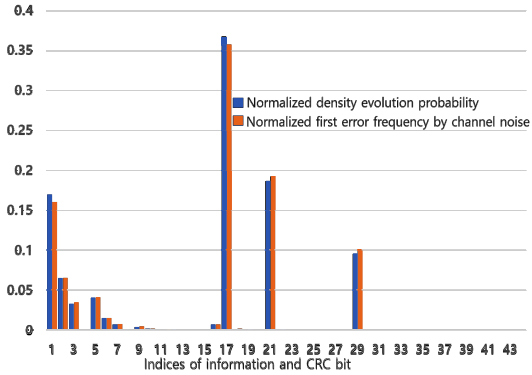


그림 1. 정보 및 CRC 비트의 정규화된 밀도 진화 확률과 채널 잡음에 의한 첫 번째 오류 빈도수의 정규화 분포
Fig. 1. Normalized density evolution probability and distribution of first error frequency by channel noise of information and CRC bit indices

을 차례대로 한 번에 한 개씩 반전시키고 재정렬된 모든 CRC 비트가 CRC 검사에 성공할 때까지 SC 복호를 최대 T 만큼 반복 수행한다. ISCF 복호 알고리즘의 순서는 알고리즘 1을 통해서 확인할 수 있다.

그림 1은 ISCF 복호의 두 번째 과정에서 사용되는 F_j 를 정의하기 위해서 오프라인에서 밀도 진화 시뮬레이션을 사용한 근거이다. 그림 1을 통해 정보 비트마다 채널 잡음에 의해 처음으로 잘못 추정된 경우의 빈도수의 분포도와 밀도 진화 시뮬레이션을 통해서 확인된 오류 확률 분포도가 유사하다는 것을 확인할 수 있다. 그러므로 후보군 선택 기준으로 시뮬레이션 결과를 빠르게 확인할 수 있는 밀도 진화 확률을 선택하고, 확률이 높은 순서대로 정렬된 정보 비트들이 F_j 가 된다.

ISCF 복호는 두 가지 측면에서 계산 복잡도를 줄인다. 첫 번째는 F_j 가 오프라인에서의 밀도 진화 시뮬레이션 통해서 결정되기 때문에 후보군 선택을 위한 추가적인 정렬 및 비교 연산이 필요하지 않다는 점이다. 두 번째는 재정렬된 CRC 비트들이 체크포인트로 활용하기 때문에 모든 비트를 추정하지 않더라도 오류 여부를 확인할 수 있다는 점이다. 이 두 가지 측면 때문에 ISCF 복호가 SCF-FIS 복호보다 더 낮은 계산 복잡도를 가질 수 있다.

IV. 시뮬레이션 결과

ISCF 복호의 시뮬레이션과 오프라인 시뮬레이션 (목표 $E_b/N_0=5.5\text{dB}$)은 표 1의 환경에서 진행한다.

ISCF 복호의 오류 정정 성능은 그림 2를 통해서 확

표 1. ISCF 복호의 시뮬레이션 환경
Table 1. Simulation parameters of the ICSF decoding

Channel	AWGN
Code length (N)	64
Code rate (R=K/N)	0.5 = 32/64
Modulation type	BPSK
CRC polynomial(g(x))	$x^{12}+x^9+x^8+x^3+x^2+x+1$ [6]
CRC length (C)	12

인할 수 있다. 그림 2에서 표기된 SC-Oracle 복호의 역할은 SCF 복호의 성능 한계치이며, 이는 채널 잡음에 의해 처음으로 잘못 추정된 비트의 위치를 수신기에서 완벽하게 알고 있는 것으로 가정한다. ISCF 및 SCF-FIS 복호의 오류 정정 성능은 최대 후보군 반전 개수(T)가 증가할수록 향상되며, $E_b/N_0=5.5\text{dB}$ 에서 서로의 성능이 유사해짐을 확인할 수 있다.

$E_b/N_0=5.5\text{dB}$ 에서 발생한 오류 프레임에 대해서만 ISCF 및 SCF-FIS 복호의 계산 복잡도가 계산된다. 이때, 단 한 번의 SC 복호 수행으로 모든 입력 비트가 추정된 경우의 계산 복잡도를 '1'로 취급한다.

하나의 오류 프레임에 대한 SCF-FIS 복호의 계산 복잡도는 처음 SC 복호 외에 r 번의 ($1 \leq r \leq T$) 비트 반전 및 SC 복호가 추가로 수행되었다고 할 때 $r+1$ 이 된다. 이를 모든 오류 프레임에 대해 평균을 취하면 SCF-FIS 복호의 계산 복잡도가 된다.

하나의 오류 프레임에 대한 ISCF 복호의 계산 복잡도는 처음 SC 복호에서 처음으로 CRC 검사가 실패하는 CRC 비트의 위치를 s_0 라고 하고 n 번째

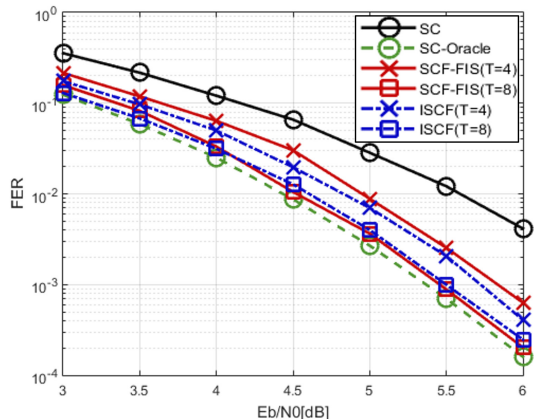


그림 2. SC, SC-Oracle, SCF-FIS 및 ISCF 복호의 오류 정정 성능 ($T=4, T=8$)
Fig. 2. The error-correction performance of SC, SC-Oracle, SCF-FIS and ISCF decoding. ($T=4, T=8$)

표 2. ISCF 및 SCF-FIS 복호의 계산 복잡도
 Table 2. The computational complexity of ISCF and SCF-FIS decoding

# of bits to flip	Decoding		
	T=4	T=8	T=12
SCF-FIS	3.3503	3.9546	4.2508
ISCF	2.1617	2.3279	2.4371
Reduction ratio	35.47%	41.13%	42.67%

($1 \leq n \leq r$) 추가 복호에서 CRC 검사가 실패하는 CRC 비트의 위치를 S_n 라고 할 때 중단한 비트 이후의 복호는 수행하지 않으므로 다음과 같다.

$$\sum_{n=0}^r \frac{S_n}{K+C} \quad (3)$$

이 값들을 모든 오류 프레임에 대하여 평균을 취하면 ISCF 복호의 계산 복잡도가 된다.

표 2를 통해, ISCF 및 SCF-FIS 복호의 계산 복잡도와 둘 사이의 계산 복잡도 감소 비율을 확인할 수 있다. T 가 커질수록 ISCF와 SCF-FIS 복호 사이의 계산 복잡도 감소 비율이 증가하고, T 가 4일 때, 계산 복잡도 감소 비율이 최소가 됨을 알 수 있다. 또한, T 가 증가할수록 각 각의 계산 복잡도가 증가하지만 큰 폭으로 증가하지 않음을 알 수 있다.

References

- [1] E. Arıkan, "Channel polarization: A method for constructing capacity achieving codes for symmetric binary input memoryless channels," *IEEE Trans. Inf. Theory*, vol. 55, no. 7, pp. 3051-3073, Jul. 2009.
- [2] J.-H. Kim, Y. J. Choi, S.-H. Kim, and H. Park, "A coding-sequence-based puncturing scheme for length compatible polar codes," *J. KICS*, vol. 42, no. 11, pp. 2093-2096, Nov. 2017.
- [3] I. Tal and A. Vary, "List decoding for polar codes," *IEEE Trans. Inf. Theory*, vol. 66, no. 5, pp. 2213-2226, May 2015.
- [4] O. Afisiadis, A. balatsoukas-Stimming, and A. Burg, "A low-complexity improved successive cancellation decoder for polar codes," in *Proc. Asilomar Conf. Sig. Syst. Comput.*, pp. 2116-2120, Pacific Grove, CA, Nov. 2014.
- [5] C. Condo, F. Ercan, and W. J. Gross, "Improved successive cancellation flip decoding of polar codes based on error distribution," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Barcelona, Spain, Apr. 2018.
- [6] "CRC Polynomial Zoo," Available: [Online]. <https://users.ece.cmu.edu/~koopman/crc/crc12.html>